

1 1. A method comprising:
2 initiating a direct memory access; and
3 successively transferring data from linked
4 buffers in a first processor system to linked buffers in a
5 second processor system.

1 2. The method of claim 1 wherein successively
2 transferring data from linked buffers includes successively
3 transferring data from buffers arranged in a linked list on
4 a first processor system to buffers arranged in a linked
5 list on a second processor system.

1 3. The method of claim 2 including providing
2 descriptors that indicate the status of each of said
3 buffers.

1 4. The method of claim 3 including providing flags
2 that indicate whether a buffer is empty or full.

1 5. The method of claim 1 including transferring data
2 between buffers in a cellular telephone.

1 6. The method of claim 5 including transferring data
2 between a first processor system that includes a baseband
3 processor and a second processor system that includes a
4 multimedia processor.

1 7. The method of claim 1 including determining the
2 status of a buffer to which data is to be transferred
3 before transferring the data.

1 8. The method of claim 7 including determining if a
2 buffer from which data is to be transferred is empty and if
3 so, automatically filling the buffer with data.

1 9. The method of claim 8 including generating an
2 interrupt when a buffer is empty and data is to be
3 transferred from the buffer, intercepting the interrupt,
4 and automatically filling the buffer.

1 10. The method of claim 9 including determining
2 whether a buffer that is to receive data is full and if the
3 buffer is full, automatically generating an interrupt,
4 intercepting the interrupt, and automatically emptying the
5 buffer.

1 11. An article comprising a medium storing
2 instructions that enable a processor-based system to:
3 initiate a direct memory access; and
4 successively transfer data from linked buffers in
5 a first processor system to linked buffers in a second
6 processor system.

1 12. The article of claim 11 further storing
2 instructions that enable the processor-based system to
3 successively transfer data from linked buffers arranged in
4 a linked list on a first processor system to buffers
5 arranged in a linked list on a second processor system.

1 13. The article of claim 12 further storing
2 instructions that enable the processor-based system to
3 provide descriptors that indicate the status of each of
4 said buffers.

1 14. The article of claim 13 further storing
2 instructions that enable the processor-based system to
3 provide flags that indicate whether a buffer is empty or
4 full.

1 15. The article of claim 11 further storing
2 instructions that enable the processor-based system to
3 transfer data between buffers in a cellular telephone.

1 16. The article of claim 15 further storing
2 instructions that enable the processor-based system to
3 transfer data between a first processor system that
4 includes a baseband processor and a second processor system
5 that includes a multimedia processor.

1 17. The article of claim 11 further storing
2 instructions that enable the processor-based system to
3 determine the status of a buffer to which data is to be
4 transferred before transferring the data.

1 18. The article of claim 17 further storing
2 instructions that enable the processor-based system to
3 determine if a buffer from which data is to be transferred
4 is empty and if so, automatically fill the buffer with
5 data.

1 19. The article of claim 18 further storing
2 instructions that enable the processor-based system to
3 generate an interrupt when a buffer is empty and data is to
4 be transferred from the buffer, intercept the interrupt,
5 and automatically fill the buffer.

1 20. The article of claim 19 further storing
2 instructions that enable the processor-based system to
3 determine whether a buffer that is to receive data is full
4 and if the buffer is full, automatically generate an
5 interrupt, intercept the interrupt, and automatically empty
6 the buffer.

1 21. A system comprising:
2 a processor; and
3 a storage coupled to said processor storing
4 instructions that enable the processor to:
5 initiate a direct memory access; and
6 successively transfer data from linked
7 buffers in a first processor system to linked buffers in a
8 second processor system.

1 22. The system of claim 21 wherein said storage
2 stores instructions that enable the processor to
3 successively transfer data from linked buffers arranged in
4 a linked list on a first processor system to buffers
5 arranged in a linked list on a second processor system.

1 23. The system of claim 22 wherein said storage
2 stores instructions that enable the processor to provide
3 descriptors that indicate the status of each of said
4 buffers.

1 24. The system of claim 23 wherein said storage
2 stores instructions that enable the processor to provide
3 flags that indicate whether a buffer is empty or full.

1 25. The system of claim 21 wherein said system is a
2 cellular telephone.

1 26. The system of claim 25 wherein said processor is
2 a baseband processor, said system further including a
3 multimedia processor.

1 27. The system of claim 21 wherein said storage
2 stores instructions that enable the processor to determine
3 the status of a buffer to which data is to be transferred
4 before transferring the data.

1 28. The system of claim 27 wherein said storage
2 stores instructions that enable the processor to determine
3 if a buffer from which data is to be transferred is empty
4 and if so, automatically fill the buffer with data.

1 29. The system of claim 28 wherein said storage
2 stores instructions that enable the processor to generate
3 an interrupt when a buffer is empty and data is to be
4 transferred from the buffer, intercept the interrupt, and
5 automatically fill the buffer.

1 30. The system of claim 29 wherein said storage
2 stores instructions that enable the processor to determine
3 whether a buffer that is to receive data is full and if the
4 buffer is full, automatically generate an interrupt,

5 intercept the interrupt, and automatically empty the
6 buffer.